LISTING OF THE CLAIMS

1-12. (canceled)

13. (previously presented) A data processing apparatus performing predetermined data processing in accordance with instruction codes read from a program memory storing a program, the data processing apparatus comprising:

a debugging circuit having a plurality of bug address setting registers and a plurality of coincidence detecting circuits,

one of said plurality of bug address setting registers holding one of a plurality of bug addresses that show the start of a buggy part of said program stored in said program memory,

one of said plurality of coincidence detecting circuits comparing a program address for reading instruction codes from said program memory with said one of said plurality of bug addresses held in said one of said plurality of bug address setting registers, said one of said plurality of coincidence detecting circuits outputting one of a plurality of coincidence signals when said program address and said one of said plurality of bug addresses coincide,

another of said plurality of bug address setting registers holding another of said plurality of bug addresses that show the start of another buggy part of the program stored in the program memory,

another of said plurality of coincidence detecting circuits comparing said program address for reading instruction codes from said program memory with said another of said plurality of bug addresses held in said another of said plurality of bug address setting registers, said another of said plurality of coincidence detecting

circuits outputting another of said plurality of coincidence signals when said program address and said another of said plurality of bug addresses coincide; and

a central processing unit receiving said plurality of coincidence signals, wherein said central processing unit:

executes one of a plurality of debugging programs stored within random access memory when said one of said plurality of coincidence signals indicates a coincidence of said program address and said one of said plurality of bug addresses,

executes another of said plurality of debugging programs stored within said random access memory when said another of said plurality of coincidence signals indicates a coincidence of said program address and said another of said plurality of bug addresses, and

executes said program stored within said program memory when said plurality of coincidence signals indicates a non-coincidence of said program address and any of said plurality of bug addresses.

14. (previously presented) A data processing apparatus as set forth in claim 13, wherein an interrupt request for said central processing unit is generated when any of said plurality of coincidence signals indicates a coincidence of said program address and any of said plurality of bug addresses.

15. (previously presented) A data processing apparatus as set forth in claim 14, wherein, when said one of said plurality of coincidence signals indicates said coincidence of said program address and said one of said plurality of bug addresses, said central processing unit:

suspends execution of said program stored within said program memory after receiving said interrupt request,

processes an instruction stored within said random access memory at said one of said plurality of bug addresses to begin execution of said one of a plurality of debugging programs after suspending execution of said program,

suspends execution of said one of a plurality of debugging programs by processing an instruction residing within said one of a plurality of debugging programs that has a return address, and

resumes execution of said program stored within said program memory by processing an instruction residing within said program memory at said return address.

16. (previously presented) A data processing apparatus as set forth in claim 14, wherein, when said another of said plurality of coincidence signals indicates said coincidence of said program address and said another of said plurality of bug addresses, said central processing unit:

suspends execution of said program stored within said program memory after receiving said interrupt request,

processes an instruction stored within said random access memory at said another of said plurality of bug addresses to begin execution of said another of a plurality of debugging programs after suspending execution of said program,

suspends execution of said another of a plurality of debugging programs by processing an instruction residing within said another of a plurality of debugging programs that has a return address, and

resumes execution of said program stored within said program memory by processing an instruction residing within said program memory at said return address.

17. (previously presented) A data processing apparatus as set forth in claim 14, wherein said plurality of bug addresses is stored within said random access memory.

18. (previously presented) A data processing apparatus as set forth in claim 13, wherein said plurality of coincidence signals is a plurality of interrupt request signals.

19. (previously presented) A data processing apparatus as set forth in claim 13, wherein said central processing unit receives said plurality of coincidence signals as separate interrupt requests.

20. (previously presented) A data processing apparatus as set forth in claim 13, wherein said central processing unit receives said plurality of coincidence signals as a single interrupt request.

21. (previously presented) A data processing apparatus as set forth in claim 20, wherein said plurality of coincidence signals are logically AND'ed together and input to said central processing unit as an interrupt request signal.

22. (previously presented) A data processing apparatus as set forth in claim 13, wherein said plurality of debugging programs are input during initialization into said random access memory from a source external to said data processing apparatus.

23. (previously presented) A data processing apparatus as set forth in claim 13, wherein said random access memory stores a plurality of interrupt vectors of start addresses, said start addresses identifying memory areas within said random access memory that contain said plurality of debugging programs.

24. (previously presented) A data processing apparatus as set forth in claim 13, wherein said central processing unit suspends an instruction being executed and reads an instruction code from a program address designated by a predetermined address table when said central processing unit executes any of said plurality of debugging programs stored within random access memory.

25. (previously presented) A data processing apparatus as set forth in claim 13, wherein said program memory is read only memory.